



## Accessible Forms

### ***Yes, Your Forms Can be Accessible!***

Making forms accessible can be a challenge, but by following a handful of simple rules, you can greatly reduce the difficulty of this task. And like most accessibility improvements, the end result will be more useful to all your users.

### ***Usability***

Forms should also be organized in a logical manner. Place the label for the form (e.g., First Name:) adjacent to the form element (i.e., text box, check box, radio button, menu/list).

Provide clear instructions about what information is desired. If any form elements are required, be sure to indicate so. Make sure that the order in which form elements are accessed is logical and easy.

### ***The Basics***

- Make forms input device independent
- Use JavaScript with care
- Correctly use the `<label>`, `<fieldset>`, and `<legend>` tags.
- Test for accessibility!

### ***<label>***

- Provide markup for radio buttons and checkboxes, using the `<label>` tag. If you already have clearly indicative text, go ahead and make that your label.
- When a screen reader accesses a form item that has a label associated with it, it will read the label and indicate the type of form item it is (e.g., "First Name. Text box" or "Age. Check box").
- Labels are needed for all form elements, except for buttons.

### ***<fieldset> and <legend>***

- `<fieldset>` explicitly ties related items to an intelligible description.
- Group related form elements, including radio buttons and checkboxes, using the `<fieldset>` tag.
- Provide a title for each fieldset using the `<legend>` tag.
- Fieldsets should be used when there are groups of check boxes or radio buttons. Other uses are possible - think about how your information is clustered.

### ***JavaScript***

- Allow your user control of the experience, and don't rely on a specific input device.
- Do not use JavaScript to automatically change the location of the Web browser (i.e., the `onChange` event). Allow the user to select the menu item, then select an adjacent form button to go to the page for the item they have selected.
- Always provide a button to submit forms - don't use JavaScript to automatically submit them.

## **Code Samples**

### **INPUT**

```
<label for="firstname">First Name</label>  
<input id="firstname" type="text" name="firstname">
```

### **TEXTAREA**

```
<label for="motives">Explain your motives:</label><br>  
<textarea id="motives" name="motives"></textarea>
```

### **CHECKBOX**

```
<fieldset>  
<legend>Choose a color:</legend><br>  
<input id="blue" type="checkbox" name="checkbox" value="checkbox">  
<label for="blue">Blue</label><br>  
<input id="yellow" type="checkbox" name="checkbox2" value="checkbox">  
<label for="yellow">Yellow</label>  
</fieldset>
```

### **RADIO BUTTON**

```
<fieldset>  
<legend>Choose a car:</legend><br>  
<input id="dart" type="radio" name="radio" value="dart">  
<label for="dart">Dodge Dart</label><br>  
<input id="pinto" type="radio" name="radio" value="pinto">  
<label for="pinto">Ford Pinto</label>  
</fieldset>
```

### **SELECT**

```
<label for="favcity">Which is your favorite city?</label>  
<select id="favcity" name="favcity">  
<option value="1">Amsterdam</option>  
<option value="2">Interlaken</option>  
<option value="3">Moscow</option>  
</select>
```

### **Handy Links**

Contact me at [jason\\_morningstar@unc.edu](mailto:jason_morningstar@unc.edu)

UNC's accessibility site is chock full of useful information: <http://www.unc.edu/webaccess>

WebAIM's forms tutorial, from which, along with Joe Clark's [Building Accessible Websites](#), this document borrows: <http://www.webaim.org/techniques/forms/>